



US005258855A

United States Patent [19][11] **Patent Number:** **5,258,855****Lech et al.**[45] **Date of Patent:** **Nov. 2, 1993****[54] INFORMATION PROCESSING
METHODOLOGY**

[75] **Inventors:** **Robert Lech**, Jackson; **Mitchell A. Medina**, Essex Fells; **Catherine B. Elias**, Plainsboro, all of N.J.

[73] **Assignee:** **System X, L. P.**, New York, N.Y.

[21] **Appl. No.:** **672,865**

[22] **Filed:** **Mar. 20, 1991**

[51] **Int. Cl.⁵** **H04H 1/40**

[52] **U.S. Cl.** **358/462; 358/448;**
358/453; 382/61; 382/48

[58] **Field of Search** 358/400, 401, 403, 447,
358/448, 449, 451, 452, 453, 460, 462, 463, 467,
470, 471, 474; 382/61, 48

[56] References Cited**U.S. PATENT DOCUMENTS**

| | | | |
|-----------|---------|--------------|---------|
| 4,034,343 | 7/1977 | Wilmer | 382/37 |
| 4,667,248 | 5/1987 | Kanno | 358/451 |
| 5,034,990 | 7/1991 | Klees | 382/50 |
| 5,095,445 | 3/1992 | Sekiguchi | 358/403 |
| 5,140,650 | 8/1992 | Casey et al. | 382/61 |
| 5,153,927 | 10/1992 | Yamanari | 382/61 |

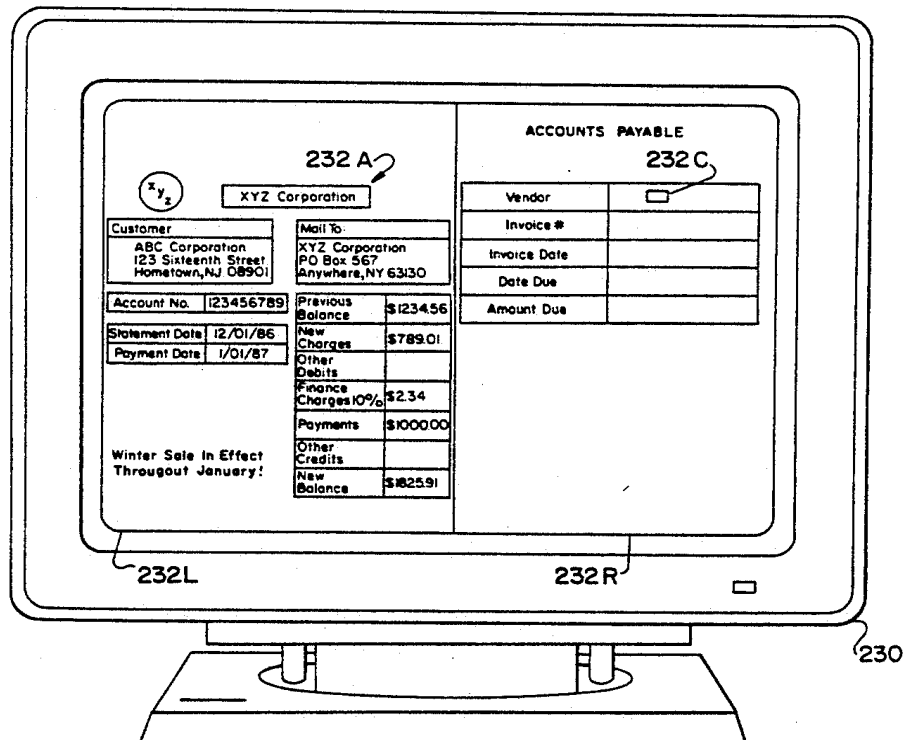
OTHER PUBLICATIONS

Que's Computer User's Dictionary 2nd Ed. Bryan Pfaffenberger (author); ©1991; p. 144.

Primary Examiner—Edward L. Coles, Sr.
Assistant Examiner—Jerome Grant, II
Attorney, Agent, or Firm—Foley & Lardner

[57] ABSTRACT

An information processing methodology gives rise to an application program interface which includes an automated digitizing unit, such as a scanner, which inputs information from a diversity of hard copy documents and stores information from the hard copy documents into a memory as stored document information. Portions of the stored document information are selected in accordance with content instructions which designate portions of the stored document information required by a particular application program. The selected stored document information is then placed into the transmission format required by a particular application program in accordance with transmission format instructions. After the information has been transmission formatted, the information is transmitted to the application program. In one operational mode, the interface interactively prompts the user to identify, on a display, portions of the hard copy documents containing information used in application programs or for storage.

77 Claims, 15 Drawing Sheets

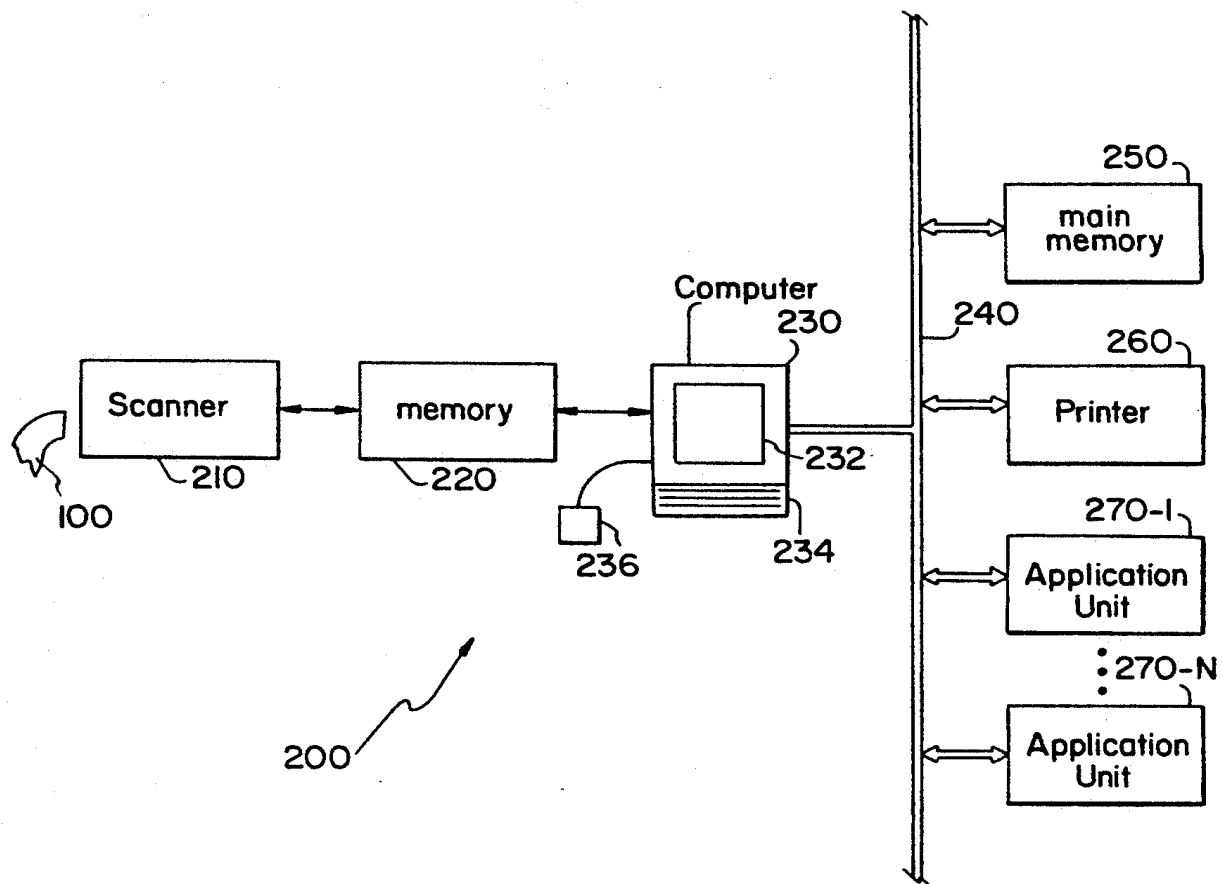
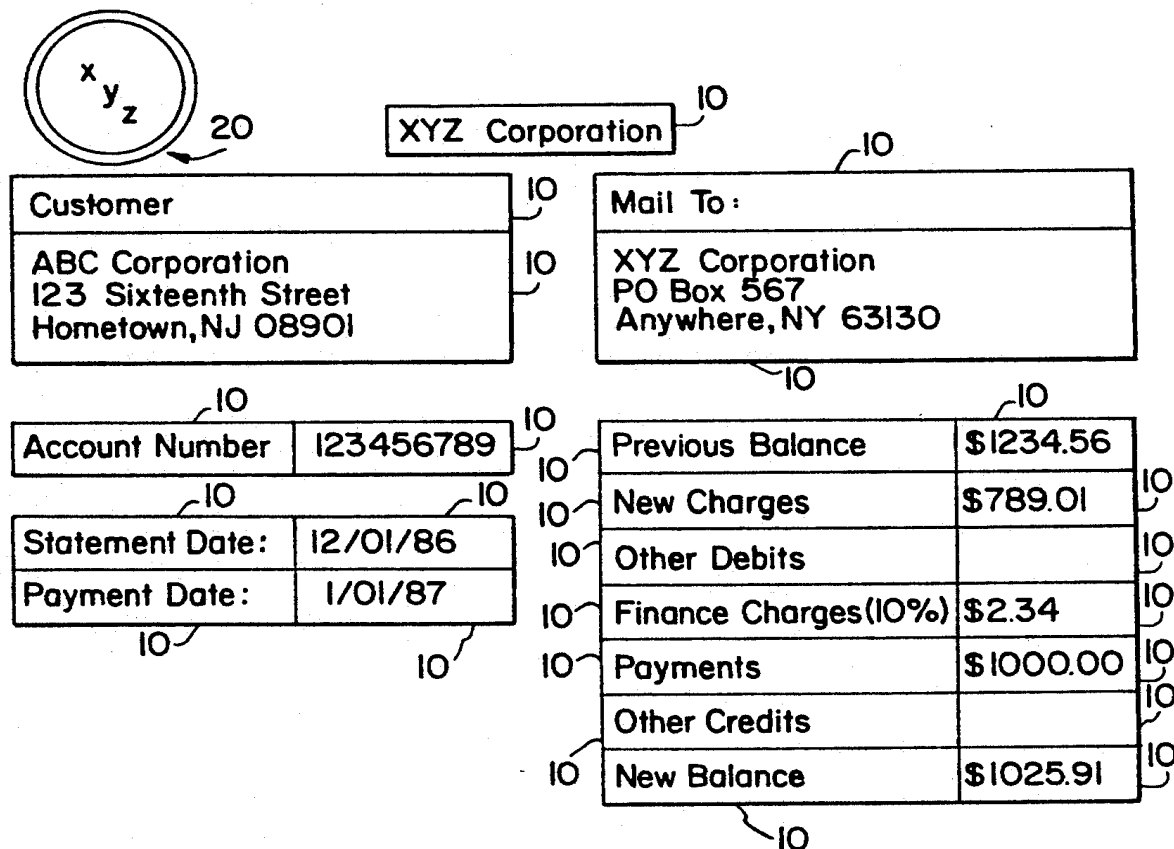


FIG. 1



The diagram illustrates a bill layout with several components:

- Logo (20):** A circular logo containing the letters 'x', 'y', and 'z'.
- XYZ Corporation (10):** A rectangular box containing the text 'XYZ Corporation'.
- Customer (10):** A rectangular box containing the text:

| |
|---|
| Customer |
| ABC Corporation 123 Sixteenth Street Hometown, NJ 08901 |
- Mail To: (10):** A rectangular box containing the text:

| |
|---|
| Mail To: |
| XYZ Corporation PO Box 567 Anywhere, NY 63130 |
- Account Number (10):** A rectangular box containing the text:

| | |
|----------------|-----------|
| Account Number | 123456789 |
|----------------|-----------|
- Statement Date: (10):** A rectangular box containing the text:

| | |
|-----------------|----------|
| Statement Date: | 12/01/86 |
|-----------------|----------|
- Payment Date: (10):** A rectangular box containing the text:

| | |
|---------------|---------|
| Payment Date: | 1/01/87 |
|---------------|---------|
- Table (10):** A table with two columns and seven rows:

| | |
|----------------------|-----------|
| Previous Balance | \$1234.56 |
| New Charges | \$789.01 |
| Other Debits | |
| Finance Charges(10%) | \$2.34 |
| Payments | \$1000.00 |
| Other Credits | |
| New Balance | \$1025.91 |

Winter Sale In Effect Throughout January!

30

FIG. 2

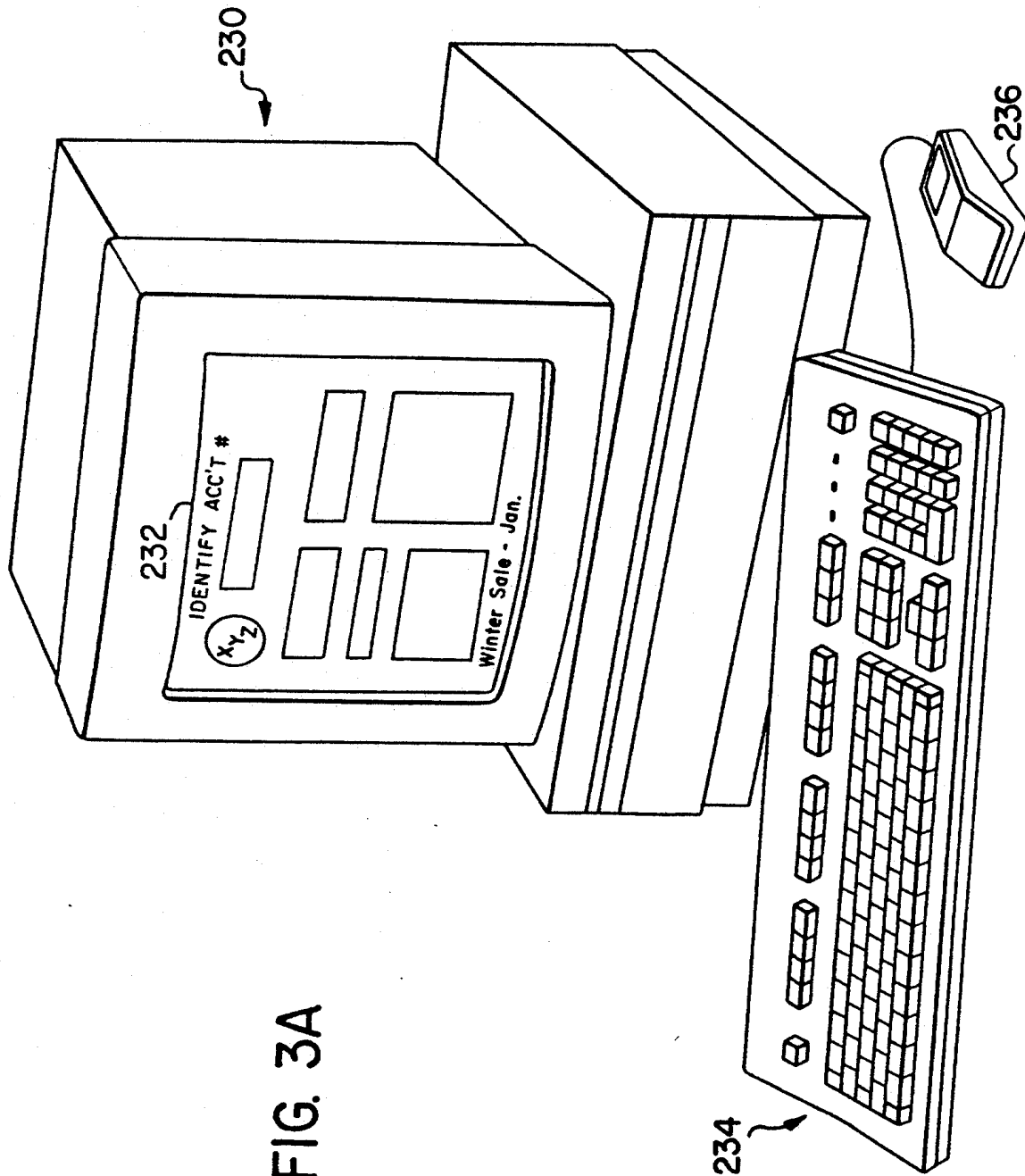


FIG. 3A

232A

Customer
ABC Corporation
123 Sixteenth Street
Hometown, NJ 08501

Mail To:
XYZ Corporation
PO Box 567
Anywhere, NY 63130

| | | | |
|----------------|-----------|----------------------|-----------|
| Account No. | 123456789 | Previous Balance | \$1234.56 |
| Statement Date | 12/01/86 | New Charges | \$789.01 |
| Payment Date | 1/01/87 | Other Debits | |
| | | Finance Charges (0%) | \$2.34 |
| | | Payments | \$1000.00 |
| | | Other Credits | |
| | | New Balance | \$825.91 |

Winter Sale In Effect
Throughout January!

232C

ACCOUNTS PAYABLE

| | | | | |
|--------|-----------|--------------|----------|------------|
| Vendor | Invoice # | Invoice Date | Date Due | Amount Due |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

FIG. 3B

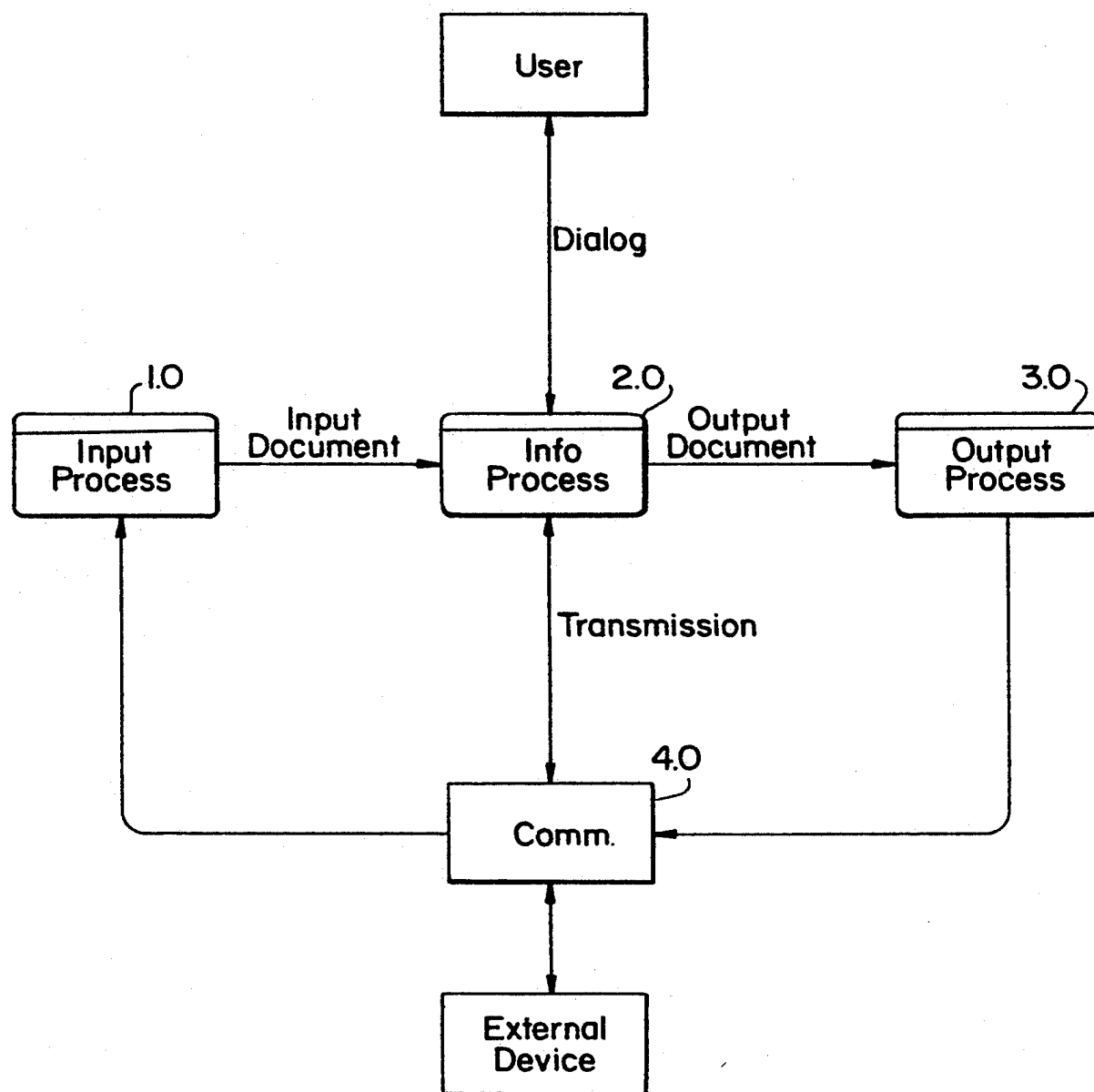


FIG. 4

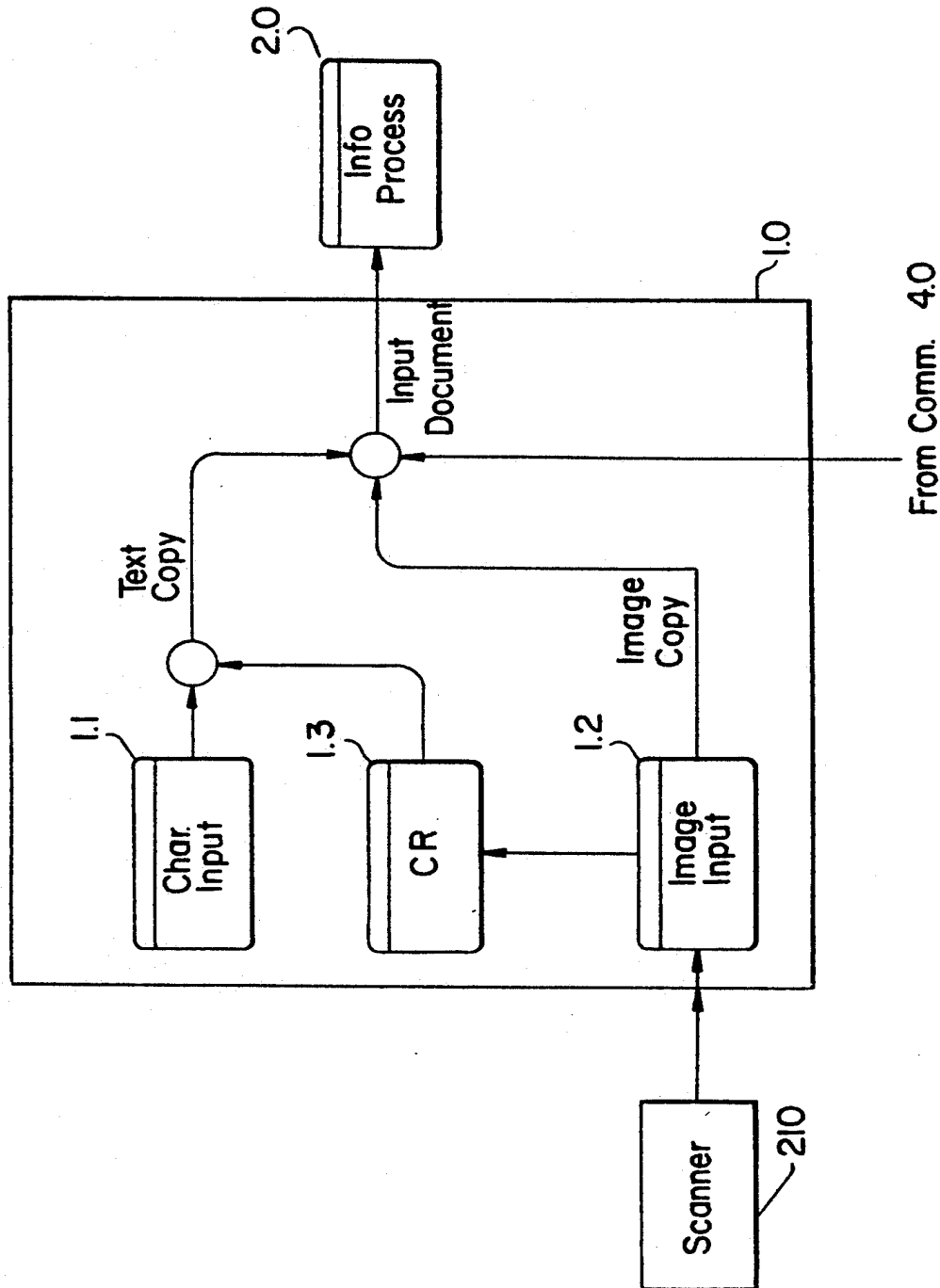
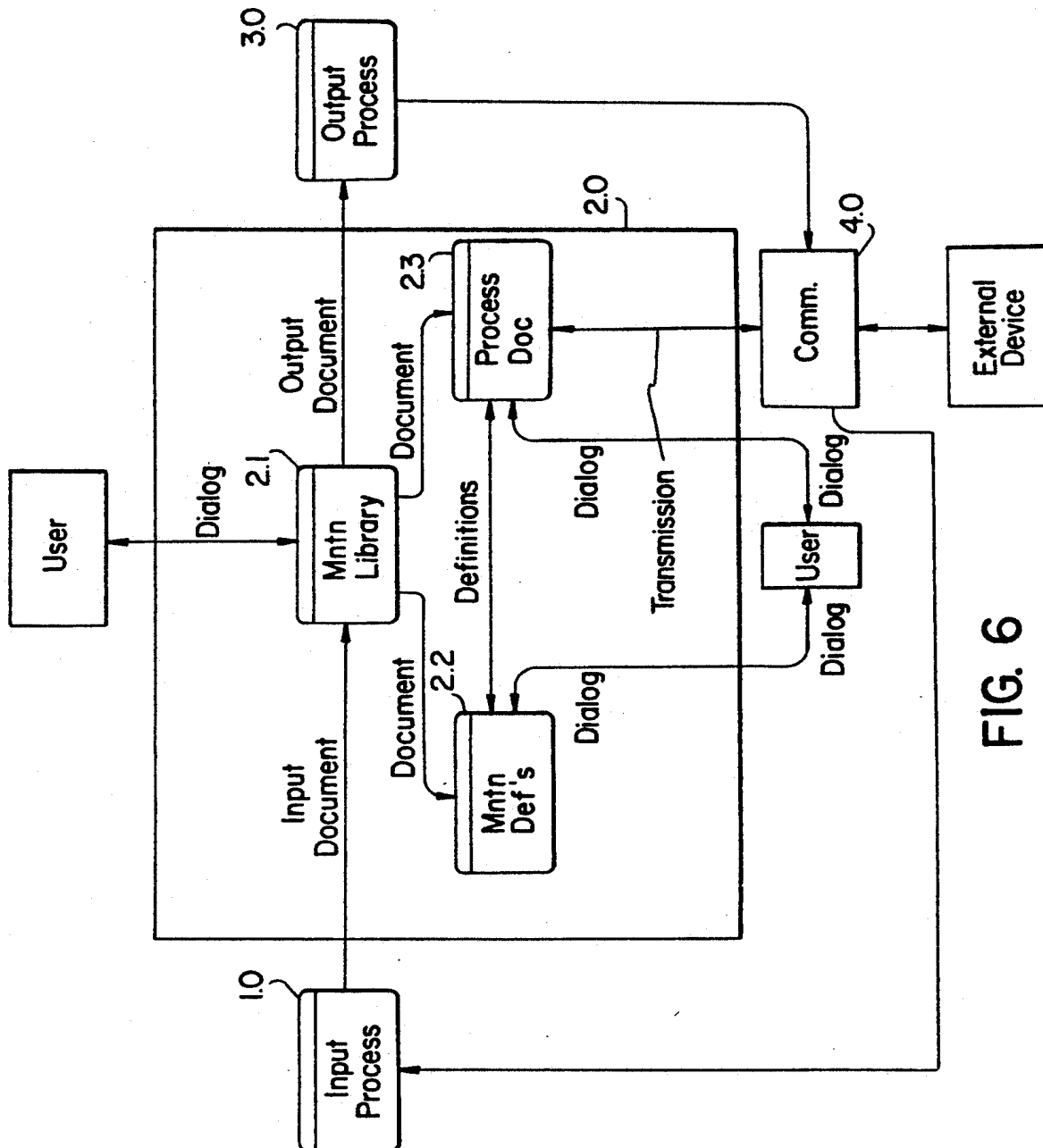
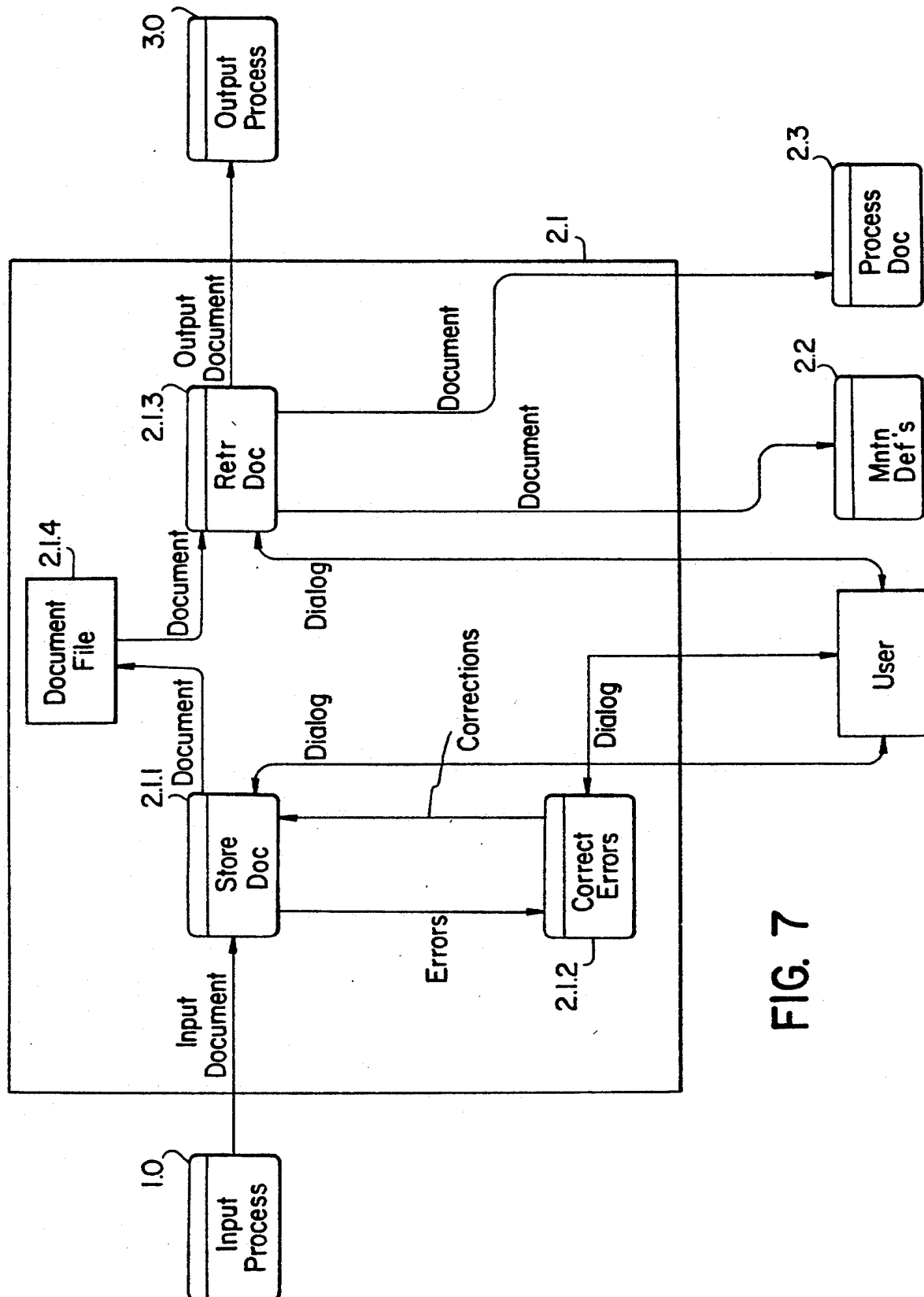
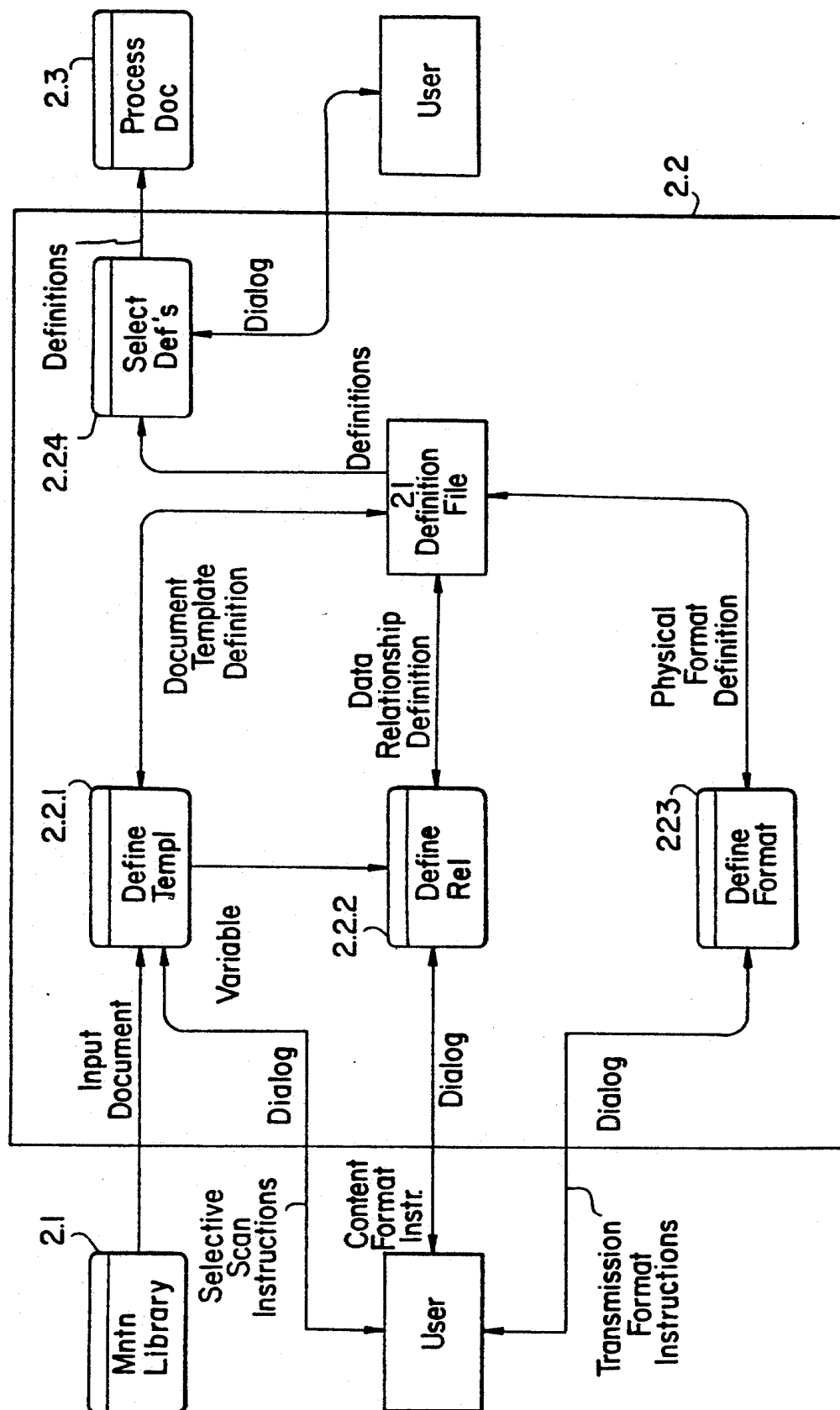


FIG. 5





8
F/G.

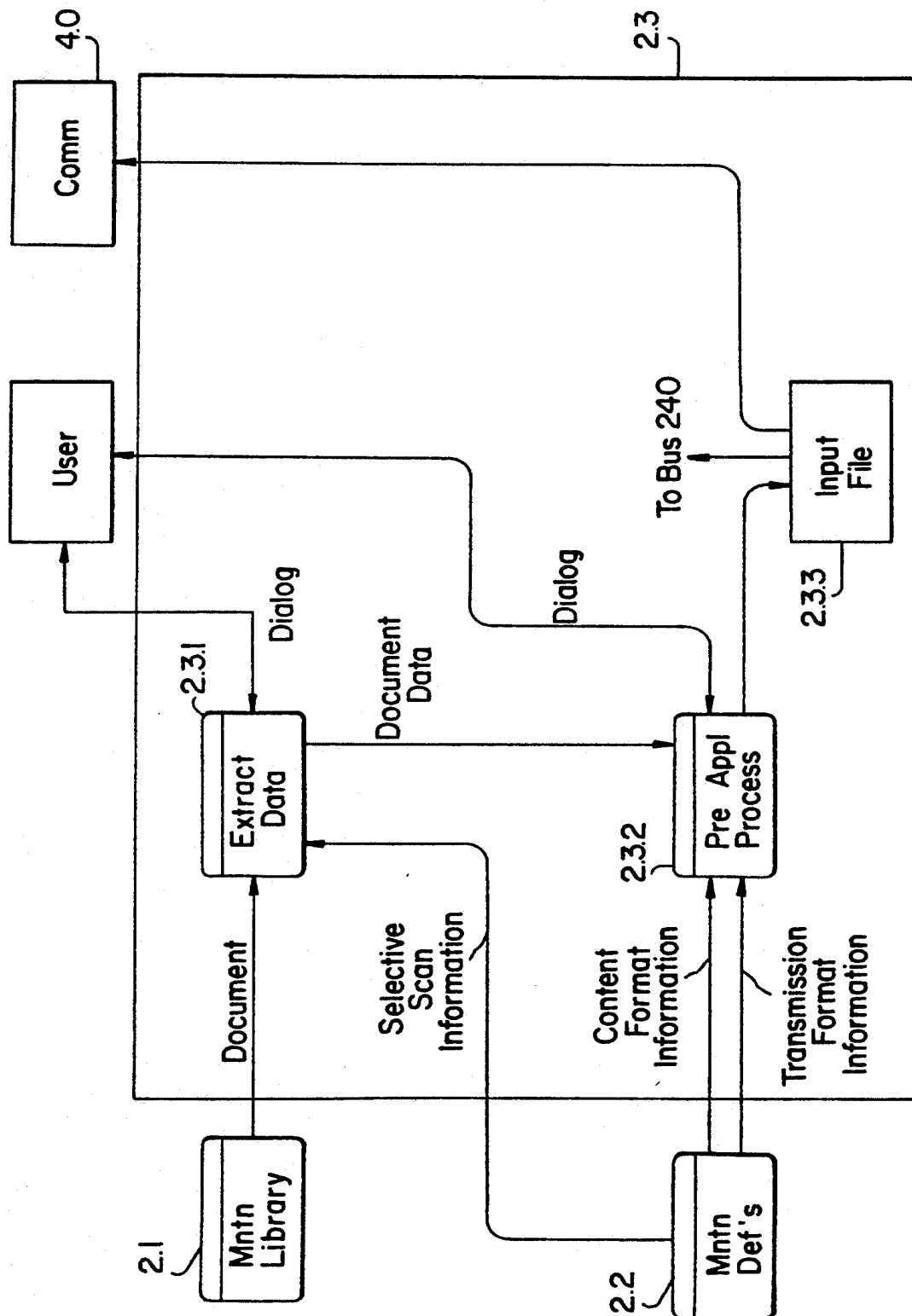


FIG. 9

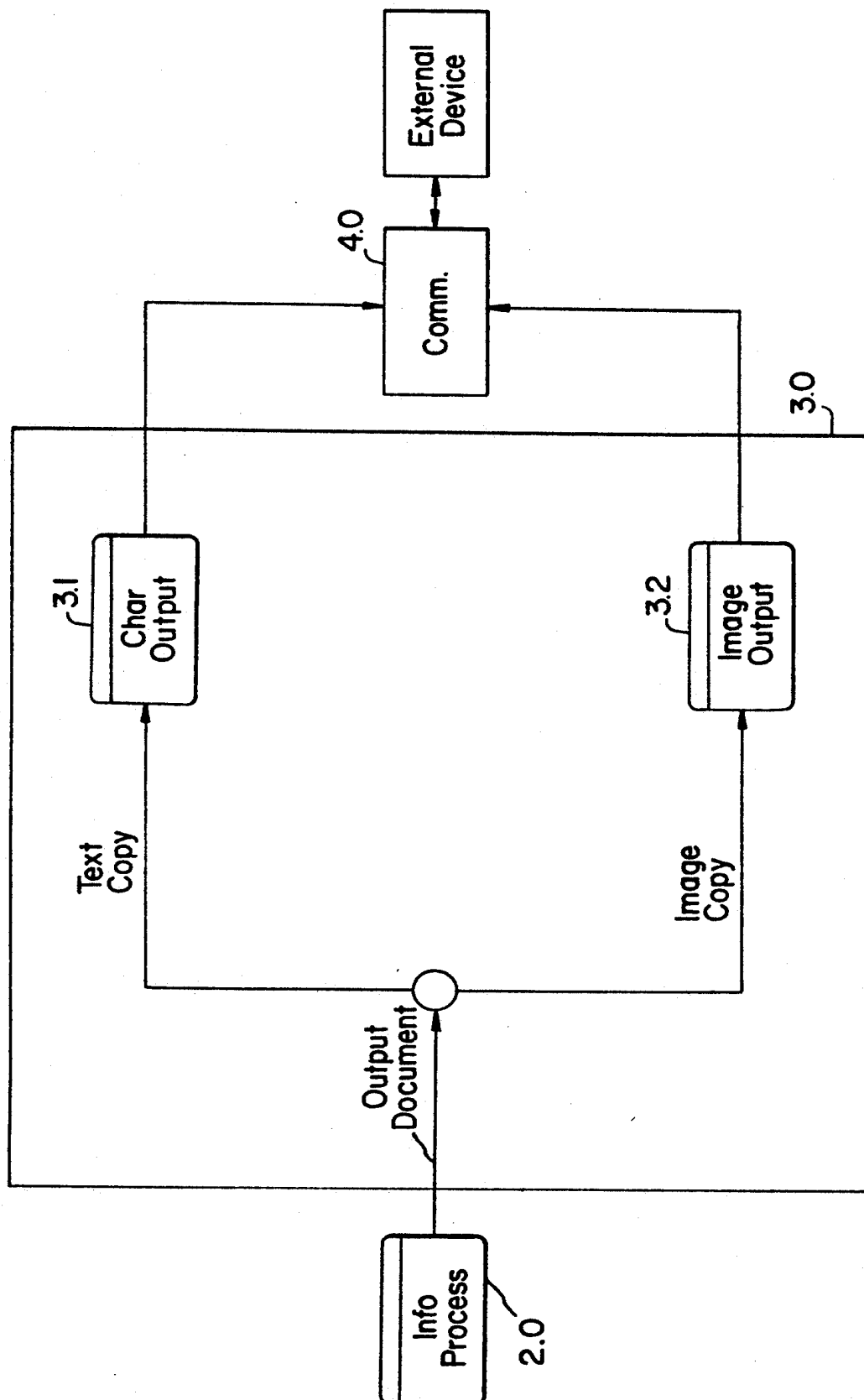


FIG. 10

| <u>Variable Name</u> | <u>Value</u> |
|----------------------|--------------------|
| Vendor | XYZ Corporation |
| Heading 2 | 2 |
| Mail To | XYZ Corporation |
| | PO Box 567 |
| | Anywhere, NY 63130 |
| Account Number | 123456789 |
| Statement Date | 12/01/86 |
| Payment Date | 1/01/87 |
| Previous Balance | \$1234.56 |
| New Charges | \$789.01 |
| Debits | |
| Finance Charges | \$2.34 |
| Payments | \$1000.00 |
| Other Credits | |
| New Balance | \$1025.91 |

FIG. II

U.S. Patent

Nov. 2, 1993

Sheet 13 of 15

5,258,855

| <u>Variable Name</u> | <u>Value</u> |
|----------------------|-----------------|
| Vendor | XYZ Corporation |
| Account Number | 123456789 |
| Statement Date | 12/01/86 |
| Payment Date | 1/01/87 |
| Previous Balance | \$1234.56 |
| New Charges | \$789.01 |
| Debits | |
| Finance Charges | \$2.34 |
| Payments | \$1000.00 |
| New Balance | \$1025.91 |

FIG. 12A

| <u>Variable Name</u> | <u>Value</u> |
|----------------------|---|
| Mail To | XYZ Corporation PO Box 567 Anywhere, NY 63130 |
| Previous Balance | \$1234.56 |

FIG. 12B

| <u>Variable Name</u> | <u>Value</u> |
|----------------------|---|
| Mail To | XYZ Corporation PO Box 567 Anywhere, NY 63130 |
| Previous Balance | \$1234.56 |

FIG. 12C

```

>1>1>1>"XYZ Corporation"
>2>2>25>+123456789>
>3>2>1>D12/O1/86>
>4>2>11>D12/I5/86>
>5>2>21>D01/O1/87>
>6>10>25>$1234.56
>7>11>25>$789.01>
>8>13>25>$2.34>
>9>14>25>$1000.00>
>10>16>25>$1025.91>

```

FIG. 13A

```

>1>1>1>$1234.56>
>2>2>1>"XYZ Corporation
>3>3>1>PO Box 567"
>4>4>1>"Anywhere,NY 63130"

```

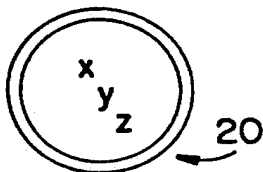
FIG. 13B

```

/1/1/1/$1234.56//
/2/2/1/*XYZ Corporation*
/3/3/1/*PO Box 567*
/4/4/1/*Anywhere,NY 63130*

```

FIG. 13C



XYZ Corporation

Customer

ABC Corporation
123 Sixteenth Street
Hometown, NJ 08901

Mail To:

▲XYZ Corporation
PO Box 567
Anywhere NY, 63130▲

Account Number 123456789

Previous Balance \$1234.56

Statement Date ②12/01/86②

New Charges ■\$789.01■

Payment Date 1/01/87

Other Debits

Finance Charges (10%) \$2.34

Payments \$1000.00

Other Credits

New Balance \$1025.91

Winter Sale in Effect Throughout January

30 ↗

FIG. 14

5,258,855

1

INFORMATION PROCESSING METHODOLOGY**BACKGROUND OF THE INVENTION**

The invention is directed to a system for efficiently processing information originating from hard copy documents. More specifically, the invention is directed to a hard copy document to application program interface which minimizes the need to manually process hard copy documents.

In the past, information contained on hard copy documents was manually entered into a computer via the input controller of a particular computer. The original document was then filed away for future reference. Automatic input of data was limited to the input of Magnetic Ink Character Recognition (MICR) data and to Optical Character Recognition (OCR) data. This fixed-position data was forwarded directly to a dedicated computer application specifically designed to accommodate the input format. In more recent years, typewritten text has been mechanically inputted into a computer via a text file. Examples of this latter type of system are word processors and photo-typesetters.

These conventional systems have limitations which decrease the efficiency of processing information from a hard copy document. For example, the systems discussed above are limited in their application to MICR, OCR, or typewritten data. Parsing and processing data is limited to the particular requirements of the particular computer application which requires the input data. In addition, in these conventional systems, the actual hard copy document must be retained for future reference at great expense.

In a sophisticated computer network, different users may require different portions of the information contained on a hard copy document. For example, if the hard copy document is an invoice returned with payment of a bill, the accounting department may need all of the monetary information contained on the bill while the mailroom may need only customer address information, to update a customer's address. Therefore, there is a need for a system in which specific information from a hard copy document can be selectively distributed to various users.

Another problem with conventional systems is that users, even within the same company, may require that the information extracted from a hard copy document be transmitted to a particular application program in a specific transmission format. For example, one department in a company may use a particular application program which must receive information using a particular character as a delimiter and other departments may require the information in a different format using different delimiters.

Another problem, particularly for small businesses, is that current systems can not efficiently accommodate the inputting of information from a diversity of hard copy documents. A large business which receives many forms in the same format can afford a system which inputs a high volume of information in that format into memory. For example, it is cost-effective for a bank which processes hundreds of thousands of checks a month to buy a dedicated machine which can read information off of checks having a rigidly defined, or fixed, format. However, as the diversity of forms received by a business increases relative to the number of forms that must be processed, it becomes less cost-effective to design a dedicated machine for processing each

2

type of form format. This problem is particularly significant in small businesses which may, for example, receive fifty invoices a month, all in different, non-fixed, formats. It is frequently not cost-effective for a small business to design dedicated systems for inputting information in each of these various formats. This leaves a small business with no other practical alternative than to manually input the information off of each invoice each month.

SUMMARY OF THE INVENTION

It is an object of the invention, therefore, to provide an application program interface which allows a user to select specific portions of information extracted from a diversity of hard copy documents and allows the user to direct portions of this information to several different users in accordance with the needs of the particular user.

It is also an object of the invention to provide a cost-effective system for inputting hard copy documents which can accommodate hard copy documents in a diversity of formats.

It is another object of the invention to provide an application program interface which allows a user to put information, which is to be transmitted, into a particular transmission format, based upon the needs of the receiver of the information.

It is a further object of the invention to provide an application program interface which will allow the extraction, selection, formatting, routing, and storage of information from a hard copy document in a comprehensive manner such that the hard copy document itself need not be retained.

It is another object of the invention to provide a system which reduces the amount of manual labor required to process information originating from a hard copy document.

A further object of the invention is to reduce the time required to process information originating from a hard copy document so that a higher volume of transactions involving hard copy documents can be processed.

The invention provides an application program interface which inputs a diversity of hard copy documents using an automated digitizing unit and which stores information from the hard copy documents in a memory as stored document information. Portions of the stored document information are selected in accordance with content instructions which define portions of the stored document information required by a particular application unit. Selected stored document information is then formatted into the transmission format used by the particular application program based on transmission format instructions. The transmission formatted selected stored document information is then transmitted to the particular application program. The hard copy documents may contain textual information or image information or both.

The interface operates in three different modes.

In a first mode, the interface extracts all of the information from hard copy documents and stores this information in memory. Parsing of various portions of the extracted information is performed in accordance with content instructions.

In a second mode, the user operates interactively with the interface by use of a display and an input device, such as a mouse. In this second mode, a hard copy document is inputted and displayed on the display. The

5,258,855

3

interface then prompts the user to identify the location of various information. For example, the interface can ask the user to identify the location of address information on the hard copy document. In response, the user positions the mouse to identify address information using a cursor. The identified information is then stored as address information in memory. Subsequently, the interface again prompts the user to identify other pieces of information, which are then stored in the appropriate locations in memory. This process proceeds until all of the information which is desired to be extracted off of the hard copy document is stored in memory.

In a third mode of operation, selected portions of information are extracted off of hard copy documents in accordance with predetermined location information which has been specified by the user. For example, the user can define a template which specifies the location of information on hard copy documents. Templates can be formed in conjunction with second mode operation. Alternatively, the user can instruct the interface to search hard copy documents for a particular character or symbol, located on the hard copy documents. The information desired to be extracted off of the hard copy documents is specified relative to the location of this character or symbol.

The interface can also prompt or receive from an applications program or another information processing system, required information, content instructions, and format instructions.

Other objects, features, and advantages of the invention will be apparent from the following detailed description of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be described in further detail below with reference to the accompanying drawings, in which:

FIG. 1 illustrates hardware for implementing a preferred embodiment of the instant invention;

FIG. 2 illustrates an example of a hard copy document containing information to be processed by the instant invention;

FIGS. 3A and 3B are enlarged views of the computer of FIG. 1 used to explain how the invention interactively prompts a user to identify information;

FIG. 4 is an overall data flow diagram for the FIG. 1 preferred embodiment;

FIG. 5 is a detailed input data flow diagram for the FIG. 1 preferred embodiment;

FIG. 6 is a detailed information processing data flow diagram for the FIG. 1 preferred embodiment;

FIG. 7 is a more detailed information processing data flow diagram for the maintain library module of FIG. 6;

FIG. 8 is a more detailed information processing data flow diagram for the maintain definitions module of FIG. 6;

FIG. 9 is a more detailed information processing data flow diagram for the process document module of FIG. 6;

FIG. 10 is a detailed output data flow diagram for the FIG. 1 preferred embodiment;

FIG. 11 lists data corresponding to the hard copy document of FIG. 2;

FIGS. 12A, 12B, and 12C illustrate examples of data which can be selected from the extracted data of FIG. 11 in accordance with content instructions;

FIGS. 13A, 13B, and 13C illustrate examples of the data of FIGS. 12A, 12B, and 12C formatted in accordance with various transmission format instructions to form input files; and

4

FIG. 14 illustrates another example of a hard copy document containing information to be processed by the instant invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Hardware

The invention provides an interface between information originating from a hard copy document and a computer application unit which uses the information. The computer application unit can be a particular computer application program or a device which is controlled in accordance with instructions or information from the hard copy document. The invention also allows storing a copy of the hard copy document in a memory and retrieving the copy of the hard copy document. By providing a comprehensive and integrated system which can accommodate almost all of the possible uses of information contained on a hard copy document, the instant invention allows for a paperless office.

The invention includes hardware and software necessary to extract, retrieve, and process information from the hard copy document. A copy of the actual image of the hard copy document is stored in memory. Textual information extracted from the hard copy document is also stored in memory. Textual information is information, such as alphanumeric characters, which is recognized on the hard copy document and which is stored in a form which corresponds to the particular recognized character. For example, the extracted characters can be stored in the ASCII format in an electronic memory.

The user can have all of the information extracted from the hard copy document and stored in memory. Alternatively, the interface can interactively prompt the user to identify specific pieces of information for storage. The interface can also extract specific pieces of information using a predefined template. The interface can also prompt or receive from another information processing system or an applications program desired information, content instructions, and format instructions.

The instant invention also provides for parsing information extracted from the hard copy document and for directing this parsed information to specific users or application programs as an input file.

The invention also permits the user to define the transmission format of the input file for a particular computer application unit.

FIG. 1 illustrates hardware for implementing a preferred embodiment of a hard copy document to application program interface according to the instant invention. The interface 200 processes information extracted off of hard copy document 100 and provides information to application units 270 in a form required by each particular application unit. The interface extracts information off of a hard copy document 100 utilizing a scanner 210. The scanner 210 can be any type of scanner which extracts information off of hard copy documents, for example, an Optical Reader.

The scanned information is stored in a scanner memory 220 or in main memory 250, as will be described in greater detail below. If main memory 250 or another memory is available to store the scanned information, then scanner memory 220 can be omitted.

5,258,855

5

The information from scanner memory 220 or main memory 250 is transmitted to computer 230. In the preferred embodiment, computer 230 includes a display 232, a keyboard 234, and a mouse 236. The display 232 displays an image of the hard copy document itself and/or information necessary to process the information extracted off of the hard copy document.

The computer 230 is used to select portions of the stored document information contained in memory in accordance with content instructions which define portions of the stored document information required by an application unit. These content instructions may be provided by the application program. Alternatively, the content instructions can be inputted via an input device such as a keyboard, a touch screen, a mouse, a notepad, a voice recognition device, or the like.

The computer 230 is also used to format selected stored document information into the transmission format used by an application unit based on transmission format instructions. The transmission format instructions may be provided by the application program. Alternatively, the transmission format instructions can be inputted via a keyboard, a touch screen, a mouse, a notepad, a voice recognition device, or the like.

Thus, the computer 230 is used to generate an input file for a particular application unit. The computer 230 is connected to scanner memory 220, main, or permanent, memory 250, a printer 260, and application units 270, via a bus 240. Although FIG. 1 illustrates use of a bus to connect components together, it is understood that any routing or connecting link, implemented in hardware or software or both, can be employed instead of, or in addition to, a bus. Instructions to or in the computer 230 control the main memory 250, the printer 260, the application units 270, and the bus 240. Instructions to or in computer 230 can also control exchanges of information with scanner memory 220.

When the computer 230 generates an input file for a particular document, the computer 230 can send this input file directly to an application unit or can store this input file in the main memory 250 until required by an application unit. The main memory 250 may also optionally store a copy of the image information for the hard copy document and the textual information for the hard copy document. Thus, the image information and textual information from the hard copy document can be retrieved and printed out on printer 260. In addition, image and textual information stored in scanner memory 220 or in main memory 250 can be used to form additional input files at the time of input or at a later time, based on content instructions and transmission format instructions. Thus, the invention can, at the discretion of the user, eliminate the need to retain copies of hard copy documents, permitting a paperless office.

The application units 270 include particular application programs and devices which are controlled in accordance with information contained on hard copy document 100.

FIG. 2 illustrates an example of a hard copy document 100 which contains information to be processed by the instant invention. The document illustrated in FIG. 2 is a bill from XYZ Corporation to customer ABC Corporation. FIG. 2 is only an example of a type of document that can be processed by the instant invention.

In a first operational mode, the scanner 210 stores all of the information extracted off of hard copy document 100 in the scanner memory 220 or, alternatively, in main

6

memory 250. The extracted information is stored in two forms. The actual image of the hard copy document 100 is stored as image information in the scanner memory 220. In addition, the scanner memory 220 stores textual information recognized on the hard copy document 100 by, for example, employing standard character recognition software. In the preferred embodiment, the textual information is stored in ASCII format. The scanner memory 220 can be, for example, an electronic, magnetic, or optical memory.

FIG. 3A illustrates an enlarged view of the computer 230 of FIG. 1. This view will be used to describe a second mode of operation. In this second mode of operation, the hard copy document 100 is scanned and a copy of the document 100 is displayed on display 232 of computer 230, based on the contents of information temporarily stored in scanner memory 220. After the document is displayed on display 232, the computer 230 interactively prompts the user to identify the location of specific pieces of information on the hard copy document. In the FIG. 3A illustration, this prompt message is indicated as the message beginning with the arrow.

For example, the prompt message can ask the user to identify the location of account number information on the hard copy document. The user then uses an input device, such as keyboard 234 or mouse 236 or a touch screen, notepad, voice recognition device, or other input device to position a cursor on the display to identify the location of the information requested by the prompt message. For example, the cursor could be used to define a block (which could be highlighted) containing the requested information, followed by a mouse "enter" click. In this example, the user would move the mouse to identify the location of the account number information contained on the hard copy document 100. The computer 230 then stores the information which has been identified by the user as account number information in the appropriate address or subfile or as the appropriate variable or parameter in memory. The computer then prompts the user to identify the location of other information on the hard copy document, such as, statement date information. The process proceeds until all of the desired information has been stored into the appropriate locations in memory.

FIG. 3B illustrates a variation of the second mode for interactively prompting the user for information. In FIG. 3B, the display is split into two portions. A left-hand portion 232L displays the image of the hard copy document and a right-hand portion 232R displays the required application program information. For example, in FIG. 3B, portion 232R displays a spreadsheet used by an application program. While observing the split display, the user can input instructions to associate specific pieces of information on the hard copy document (for example, the vendor name indicated by the mouse arrow 232A) with particular subfiles in memory (for example, the vendor field next to which the cursor 232C appears), using a mouse or other input device(s) or both. The split display also allows the user to generate content format instructions while observing the information required for a particular application program on the right-hand portion.

These second modes of operation are efficient for small businesses which receive a small number of a wide variety of invoices, since the user does not necessarily have to store all of the information that appears on the hard copy document. A further advantage is that data input is quicker, easier, and more accurate than with

7

5,258,855

previous keyboard methodology. In addition, by specifying the location on the hard copy document of information, the user may optionally create a template, to be described in further detail below, for each different type of invoice. This template is stored for future use when another hard copy document in the same format is received.

More specifically, instructions from computer 230 can direct the scanner 210 and scanner memory 220, and/or main memory 250, to scan and/or store only specific portions of hard copy document 100. After the interactive prompts required to obtain information for a desired application program, the unused information stored in scanner memory 220 or 250 can be erased. Further, scanning of a second identical document can be limited to only those portions of the document which contain needed information.

More specifically, in FIG. 2, the lines 10 drawn around certain portions of the document represent the areas which the user has previously identified as the portions of a document to be extracted by the scanner 210 and stored in scanner memory 220 and/or main memory 250. Since the logo 20 and the message 30 have not been identified as an area to be scanned and stored, these areas are not scanned and stored in subsequent documents. Since the user has previously associated each of the areas 10 with a specific subfile of information, e.g., the account number, the scanned information is stored in memory locations corresponding to that subfile.

Data Processing

FIGS. 4-10 illustrate the flow of data in the FIG. 1 preferred embodiment. FIG. 4 illustrates the overall data flow for the FIG. 1 preferred embodiment. The preferred embodiment includes an input process module 1.0, an information processing module 2.0, and an output processing module 3.0. The information processing module 2.0 is equipped to receive instructions from and transmit information to a user. The information processing module 2.0 can also transmit to and receive information from a remote external device through communication interface 4.0. Input process module 1.0 and output processing module 3.0 can also access communication interface 4.0. A module is implemented in hardware, software, or a combination of hardware and software. The specific implementation for a particular business application depends upon a variety of factors, for example, the relative costs of hardware and software implemented systems, the frequency with which a user will want to expand or modify the system, and the like.

FIG. 5 is a more detailed diagram of the input process module 1.0 of FIG. 4. The input process module 1.0 includes a character input module 1.1, an image input module 1.2, and, in the preferred embodiment, a character recognition device 1.3. The character input module inputs textual information, such as alphanumeric characters, from an input device such as keyboard 234. The image input module 1.2 inputs image information, for example, a digitized image of the actual appearance of hard copy document 100. Textual information can include textual input from an input device such as keyboard 234 and textual information extracted from the document by character recognition device 1.3. Both types of information comprise an input document which is transmitted to information processing module 2.0. In the FIG. 1 preferred embodiment, the processing per-

8

formed by input process module 1.0 occurs in scanner memory 220, computer 230, and main memory 250.

FIG. 6 illustrates information processing data flow for the FIG. 1 preferred embodiment, that is, FIG. 6 illustrates data flow in the information processing module 2.0.

The information processing module 2.0 includes a maintain library module 2.1, to be described in further detail below in conjunction with FIG. 7, a maintain definitions module 2.2, to be described in further detail below in conjunction with FIG. 8, and a process document module 2.3 to be described in further detail below in conjunction with FIG. 9.

The information processing module 2.0 is the module which coordinates and drives the entire system. In the preferred embodiment, the information processing module 2.0 is implemented primarily by computer 230.

FIG. 7 illustrates information processing data flow in the maintain library module 2.1. The maintain library module 2.1 maintains a library of image information, for example, a digitized image representing the actual appearance of the hard copy document, and textual information of the hard copy documents for reference during processing. This library can be incorporated within scanner memory 220, main memory 250, or another independent memory, for example, a RAM disk. The maintain library module 2.1 includes a store document module 2.1.1, a correct errors module 2.1.2, a retrieve document module 2.1.3, and a document file 2.1.4. These modules operate collectively to store, retrieve, and correct document information.

The store document module 2.1.1, prior to routing the document to the document file 2.1.4, may provide information on recognition errors which may have occurred while inputting the document. For example, the store document module 2.1.1 identifies that a character contained on hard copy document 100 was not recognized. The store document module 2.1.1 also optionally causes a copy of the document and its parsing to be displayed on the display 232 for confirmation by the user. The user may utilize this opportunity to identify any errors in the displayed document and, in conjunction with the correct errors module 2.1.2, to revise the document's parsing, if necessary, prior to storage of the document in memory. The module 2.1.1 also provides a facility for the user to name a particular hard copy document for cataloging, storage, and retrieval purposes. After the document is named, the store document module 2.1.1 stores copies of the document in the document file 2.1.4.

The correct errors module 2.1.2 processes instructions from the user to correct errors identified by the store document module 2.1.1 and errors that have been spotted by the user during the confirmation process.

The retrieve document module 2.1.3 permits the user to retrieve a copy of a document previously stored in the document file 2.1.4. As described above, long-term storage is provided by main memory 250, if necessary.

FIG. 8 illustrates a more detailed information processing data flow diagram for the maintain definitions module 2.2 of FIG. 6. The maintain definitions module 2.2 allows the user to define system and document parameters and maintains the definitions of these system and document parameters. The maintain definitions module 2.2 includes a define template module 2.2.1 which allows the user to specify the location of information on the document. This information provided by the user defines a template which is used to extract

information off the document and to associate the extracted information with a particular variable or subfile. These templates are illustrated by boxes 10 in the FIG. 2 example of a hard copy document. The maintain definitions module 2.2 can also access templates previously defined by the user and stored in main memory 250. Templates can also be provided as part of software packages developed by program developers.

The maintain definitions module 2.2 also includes a define relationships module 2.2.2. The define relationships module 2.2.2 allows the user to define data relationships, or logical relationships, between pieces of information extracted from the hard copy document. These pieces of information are then used to generate an input file for a selected computer application unit. The user defines these relationships by content instructions. Alternatively, content instructions to define relationships can be provided by application software. If the user provides these content instructions, the content instructions are inputted via keyboard 234 or via another input device such as a notepad, a voice recognition device, or the like. Examples of content instructions, data, and logical relationships will be described in further detail in conjunction with FIGS. 11 and 12A, 12B, and 12C.

The maintain definitions module 2.2 also includes a define format module 2.2.3. The define format module 2.2.3 allows the user to define transmission formats for an input file which is then transmitted to a selected computer application unit. Selection of the transmission format of the input file is accomplished by the user through use of transmission format instructions. Alternatively, the applications software itself can generate its own transmission format instructions. When the user must specify transmission format instructions, the transmission format instructions are inputted via keyboard 234 or via another input device such as a notepad, a voice recognition device, or the like. A further description of various transmission formats will be provided below in conjunction with FIGS. 12A, 12B, 12C, 13A, 13B, and 13C.

A select definitions module 2.2.4 is also included in the maintain definitions module 2.2. The select definitions module 2.2.4 allows the user to store and select a set of definitions to be used for processing the document. The definitions identify pieces of information on the document by, for example, absolute location, variable location, or relative location, or by proximity to key words and/or symbols. These definitions are described in further detail below by way of an illustrative example.

FIG. 9 illustrates a more detailed information processing data flow diagram for the process document module 2.3. The process document module 2.3 processes the document after the document has been stored in the system. The process document module 2.3 gathers the appropriate information which has been stored, and creates input file(s) 2.3.3 for the selected application unit. The process document module 2.3 then transmits the input file(s) via bus 240 and/or communication interface 4.0 to an application unit 270, an output device such as printer 260, or to main memory 250.

The process document module 2.3 includes an extract data module 2.3.1. This module extracts data off of the document in accordance with the user's instructions, for example, the user-defined template, or through the interactive mode.

The process document module 2.3 also includes a preapplication process module 2.3.2 which gathers and associates information extracted from the document in accordance with content instructions. This module prompts the user for any additional information required to satisfy the relationships defined by the content instructions. The preapplication process module 2.3.2 also places the selected information into the transmission format defined by the transmission format instructions.

The preapplication process module 2.3.2 also generates the input file 2.3.3 for the selected application in accordance with the appropriate instructions. The input file 2.3.3 is then transmitted to bus 240 and/or communication interface 4.0 for transmission to a particular application unit 270.

FIG. 10 illustrates a detailed output data flow diagram for output module 3.0. Output module 3.0 outputs a textual and/or image copy of the document. In the FIG. 1 preferred embodiment, output module 3.0 is implemented by printer 260, associated software, and associated interface circuitry.

Operation

Examples of operation of a preferred embodiment will now be described.

The user enters the system by providing instructions to the information processing module 2.0. The user then instructs the information processing module 2.0 to conduct maintain library processing, maintain definitions processing, or process document processing.

If the user selects maintain library processing, the user then provides instructions to maintain or modify the document library through the maintain library module 2.1. For example, the user can direct the inputting and storage of a hard copy document 100 or can retrieve and output a document. The user requests inputting of a document through the store document module 2.1.1. The system then prompts the user to specify a storage location for the inputted document. The document is then read-in by the input process module 1.0. A textual copy and/or an image copy are stored into the document file 2.1.4. Errors which have occurred during inputting are identified and corrected by the correct errors module 2.1.2 and the user. The corrections are reflected in the document information stored in document file 2.1.4.

The retrieve document module 2.1.3 is used to retrieve and output a document. The system prompts the user to specify the storage location of a document and the type of document copy, for example, a textual or an image copy, to be outputted. The document is then outputted by the output process module 3.0.

If the user initially selected maintain definitions processing, the user would instruct the system to maintain and/or modify parameter definitions through the maintain definitions module 2.2. For example, the user can define and maintain a document template for extracting selected portions of information off of the hard copy document. The user can use the template to extract selected portions of information off of the hard copy document when the document is originally inputted, or alternatively, the user can use the template to identify selected portions of information for extraction off of an image copy of the document. In creating the template, the user identifies pieces of information on the document to be extracted and assigns a variable name, or subfile, to each piece of data.

5,258,855

11

The location of data to be extracted can be defined in a number of ways other than by use of a template. For example, the user can designate the absolute location of information on the document with respect to a grid overlaid on the document, e.g., always on line 3, starting in column i. The user can also identify information by specifying the relative location of information to be extracted, e.g., always two lines below the piece of data named "salutation", starting in column 3. The user can also specify the location of information to be extracted by variable location specification. For example, if the hard copy document is a letter, the module would conduct a key word search for the term "Dear Sir:". Whenever this term "Dear Sir:" is located, this piece of data would be associated with the variable specified by the user, for example, the variable "salutation." In addition, a defined set of conventional symbols can be used to signify certain recurring data items for the convenience of users of the instant invention. For example, a "@" symbol can be used to delineate the vendor name as follows: "@XYZ Corporation@". Other examples of the use of symbols to delineate information will be described with reference to FIG. 14.

The maintains definition module 2.2 is also used to maintain data relationships in accordance with content instructions and to maintain input file formats in accordance with transmission format instructions. Relationships are defined and maintained between pieces of data, specified by, for example, the names of variables, through the define relationships module 2.2.2. The names of pieces of data on the document are retrieved by, for example, the define template module 2.2.1, and are passed to the define relationships module 2.2.2. The user may then provide any additional pieces of data needed to generate an input file for a particular application program or unit, such as an input file line number. The user, the applications software, and/or instructions previously stored in memory then establishes the contents of the input file by defining relationships between pieces of data using content instructions. Specific examples of content instructions will be discussed below in conjunction with FIGS. 11, 12A, 12B, 12C, 13A, 13B, and 13C.

The user and/or the applications software defines and maintains the transmission format of the input file to be used by a particular application program or unit through the define format module 2.2.3 in accordance with transmission format instructions. This is accomplished by defining the parameters to be used by the preapplication process module 2.3.2 in generating an input file. Parameters which would typically be required to generate an input file would include the character type, e.g., text or pixel; delimiters used between pieces of data, e.g., a slash or a semicolon; end of line characters, e.g., a carriage return or a line feed; and end of file characters. Examples of transmission formats will be described in further detail below in conjunction with FIGS. 11, 12A, 12B, 12C, 13A, 13B, and 13C.

If the user initially selected process document processing, the interface will then proceed to process the document through use of the process document module 2.3. For example, the user can extract specific portions of data from an image copy of a document, can generate an input file for transmission to an application program, or can directly process information interactively with an application program.

If the user desires to extract specific portions of data from an image copy of a hard copy document which has

12

already been stored in memory, the user uses the extract data module 2.3.1 to identify a document to be processed. The document is then retrieved by the retrieve document module 2.1.3 and passed to the extract data module 2.3.1. The user can also select parameter definitions through the select definitions module 2.2.4.

The selected document template or parameter definition is passed to the extract data module 2.3.1. The extract data module 2.3.1 extracts pieces of data from the image copy of the document, as defined by the document template definition or the parameter definitions or both. This document data is then passed to preapplication process module 2.3.2.

The interface generates input file(s) 2.3.3 by use of the preapplication process module 2.3.2. The selected data relationship definition, as defined by the content instructions, and the selected record format definitions, as defined by the transmission format instructions, are passed to the preapplication process module 2.3.2. The preapplication process module 2.3.2 assembles the input file in accordance with the content instructions. The preapplication process module 2.3.2 also prompts the user for any additional pieces of data which need to be provided by the user. The input file is converted to the desired transmission format in accordance with the transmission format instructions. This physically formatted data is then stored in the input file 2.3.3.

The user can also use an application program to process information by loading the particular application program into the computer 230 rather than by sending the input file to a remote application unit 270.

An illustrative example of the processing described above will now be described.

The user inputs instructions via keyboard 234 or another input device which indicate that the user desires to input and store a document. The computer 230 then prompts the user for the name of the document. In this example, the user desires to input the document of FIG. 2 and therefore names the document "XYZ Corp. Bill Dec. 1, 1986." The computer then prompts the user to feed the hard copy document 100 into the scanner 210. The image of the hard copy document is displayed on display 232. The computer then prompts the user to identify the account number on the document. By use of the mouse 236 or other input device to position a cursor on the display, the user indicates the location of the account number. The account number is then read-in to a subfile named "Account Number." This process proceeds until all of the desired information has been read-in and stored.

In this particular example, no errors were encountered while inputting the document. The user then directs that the document be stored for future reference in a document file.

Some time later, the user desires to retrieve and output the document and to generate input files based on information from the document. The computer 230 prompts the user for the name of the document and the type of output. The user responds with "XYZ Corp. Bill Dec. 1, 1986" for a printed textual copy. The document is then retrieved from the document file and passed to the printer 260 for printing.

In order to generate an input file for a specific application program, the user selects the option to define a document template for use when each month's XYZ Corporation bill arrives. Accordingly, the user instructs the system to display a copy of an XYZ Corporation bill on the display 232. The user then identifies pieces of

5,258,855

13

data by absolute locations. That is, the user assigns specific names to information located at specific portions of the document. In this example, the user would input the following information:

Vendor-text, line 1, one line, column 1, 80 characters;
Account number-numeric, line 6, one line, column 25, 9 characters;

Statement date-date, line 9, one line, column 25, 8 characters;

Payment date-date, line 11, one line, column 25, 8 characters;

Previous balance-currency, line 7, one line, column 75, 9 characters;

New charges-currency, line 8, one line, column 75, 9 characters;

Other debits-currency, line 10, one line, column 75, 9 characters;

Finance charges-currency, line 12, one line, column 75, 9 characters;

Payments-currency, line 13, one line, column 75, 9 characters;

Other credits-currency, line 14, one line, column 75, 9 characters;

New balance-currency, line 15, one line, column 75, 9 characters.

The user also identifies data with variable locations. In this particular example, a variable location is specified as follows:

Heading 2-line, value="Mail To:"

The identification of Heading 2 as line information means that the system will search for occurrences of the character string "Mail To:" and assign the line number which contains this character string to Heading 2.

The user also identifies data by relative locations. In this example, the user identifies the following relative location:

Mail To-text, Heading 2+1, 3 lines, column 60, 25 characters per line.

The instructions above instruct the system to assign the textual information beginning on one line after Heading 2 and continuing for 3 lines, in column 60, to the Mail To subfile.

As an alternative to inputting the actual line, column, and character numbers, the user can identify desired portions of the document by blocking, or highlighting, the desired portions using the mouse or other input device. In this case, the computer converts the highlighted portions into corresponding line, column, and character numbers.

FIG. 11 lists data corresponding to the hard copy document of FIG. 2 and the associated variable or subfile names.

Next, the user desires to define data relationships in accordance with content instructions. Examples of the type of contents which can be specified by a user are illustrated in FIGS. 12A, 12B, and 12C.

In this particular example, three separate departments of ABC Corporation require information from the XYZ Corporation bill. The first department requires vendor, account number, statement date, payment date, previous balance, new charges, debits, finance charges, payments, and new balance information. The second and third departments require mail to information and previous balance information. Each of these departments have their own application program which utilizes this information.

The user employs content instructions to designate how pieces of information, which have been extracted

14

off of hard copy document 100, are directed to particular departments, that is, particular application programs. FIG. 12A illustrates the contents of the information to be transmitted to the first department. FIG. 12B illustrates the information to be transmitted to the second department. FIG. 12C illustrates the information to be transmitted to the third department. The content instructions, therefore, parse the information shown in FIG. 11 to various application programs, as shown by FIGS. 12A, 12B, and 12C. Content instructions can also be used to identify additional pieces of data which are required for the input files of the particular application programs. In this particular example, the specific application programs from the three departments all require numeric record number information, numeric horizontal position information, numeric vertical position information, and date received information. The horizontal and vertical position information is used by the application program to specify the location of the received information on a spreadsheet application program, in this example. The user may know in advance the content format required by each application program, that is, in this example, the location and type of information specified on the spreadsheet. The user may also employ the split display mode described with reference to FIG. 3B to generate content format instructions.

Using the content instructions, the user establishes the following contents for the input file corresponding to FIG. 12A:

Record number, horizontal position, vertical position, vendor;

Record number, horizontal position, vertical position, account number;

Record number, horizontal position, vertical position, statement date;

Record number, horizontal position, vertical position, date received;

Record number, horizontal position, vertical position, payment date;

Record number, horizontal position, vertical position, previous balance;

Record number, horizontal position, vertical position, new charges;

Record number, horizontal position, vertical position, finance charges;

Record number, horizontal position, vertical position, payments;

Record number, horizontal position, vertical position, new balance.

Next, transmission format instructions are employed to define the transmission format of the input file for a specific application program or unit. FIG. 13A illustrates the transmission input file corresponding to FIG. 12A. FIG. 13B illustrates the transmission input file corresponding to FIG. 12B. FIG. 13C illustrates the transmission input file corresponding to FIG. 12C. A comparison of FIGS. 12B and 12C reveals that FIGS. 12B and 12C have the same contents. However, the information illustrated in FIG. 12B is being sent to a different application program than the information in FIG. 12C. These application programs require different transmission input formats, as illustrated in FIGS. 13B and 13C. More specifically, the application program that receives the input file illustrated in FIG. 13B uses the greater than sign as a delimiter whereas the application program which receives the transmission input file shown in FIG. 13C uses a back-slash as the delimiter.

5,258,855

15

After the contents and the transmission format for the input file have been defined, and any additional information has been inputted, the input file is assembled and transmitted to the particular application program.

FIG. 14 illustrates another example of a hard copy document containing information to be processed by the instant invention. The hard copy document illustrated in FIG. 14 is first scanned and information from the hard copy document is stored into a memory. The interface 200 then identifies portions of the hard copy document corresponding to various variables by recognizing a defined set of symbols. In the FIG. 14 example, triangles delineate the mailing address, circles delineate the statement date, and squares delineate the new charges. Information from these portions of the hard copy document is stored in the corresponding memory locations or subfiles for each variable. The same set of symbols can be used to identify the same information from one document to the next. Thus, even if the physical formats of documents are not fixed from one document to the next, a diversity of hard copy documents can be processed without manually inputting data by recognition of the defined symbols.

Examples of readily available application programs are Quicken and Lotus 1, 2, 3 both of which are widely utilized in the business community. Quicken, for example is an easy-to-utilize program for writing checks and preparing business records. Payee, amount and address information may readily be transmitted from scanner memory 220 and/or main memory 250 to the Quicken application program for check writing functions and ledger keeping purposes. Lotus is a well known spreadsheet program which may process data input into specified cells once this data is placed in conventional Lotus format.

Thus, the instant invention provides an integrated and comprehensive system for handling information from a hard copy document, thus permitting a paperless office. In addition, the invention permits data, extracted off of a hard copy document, to be easily manipulated into various logical and transmission formats required by a particular application unit. The invention also provides a low cost system for inputting information from a wide variety of hard copy documents into a memory.

The foregoing description has been set forth merely to illustrate preferred embodiments of the invention and is not intended to be limiting. Modifications are possible without departing from the scope of the invention.

For example, letters, checks, forms, pictures, reports, music scores, film, and other types of hard copy documents can be processed by the invention for accounts payable/receivable accounting, inventory control, record keeping, budgeting, data base management, music transcription, forms processing, computerized art, survey and questionnaire processing, statistical data analysis, correspondence processing and other applications.

Other automated digitizing units can be used in addition to or as an alternative to use of the scanner 210 as an input unit. Any electrical, magnetic, or optical device which extracts information off of a hard copy document, thereby eliminating the need to manually input significant amounts of information from the hard copy document is suitable for use as an automated digitizing unit. In addition, information can be input by user responses and digital and analog signals generated from various devices, and from computer files from other computer systems. Suitable hardware for inputting data

16

includes a keyboard, a light pen, a mouse, a touch screen, a laser scanner, a microphone, a tablet, a disk drive, a magnetic tape drive, and a modem.

The interface 200 can also output information in forms other than a hard copy of textual or image information. For example, the interface 200 can output system responses, computer files, and digital and analog signals for transmission to other computer systems or to control systems. Suitable hardware for outputting information includes a disk drive, a magnetic tape drive, a cathode ray tube, a plasma screen, a printer, a plotter, a film developer, an amplifier, and a modem.

Since modifications of the described embodiments incorporating the spirit and substance of the invention may occur to persons skilled in the art, the scope of the invention should be limited solely with respect to the appended claims and equivalents.

What is claimed is;

1. A method of processing information from a diversity of types of hard copy documents, said method comprising the steps of:

(a) receiving output representing a diversity of types of hard copy documents from an automated digitizing unit and storing information from said diversity of types of hard copy documents into a memory, said information not fixed from one document to the next, said receiving step not preceded by scanning, via said automated digitizing unit, of a separate document containing format requirements;

(b) recognizing portions of said hard copy documents corresponding to a first data field; and

(c) storing information from said portions of said hard copy documents corresponding to said first data field into memory locations for said first data field.

2. A method as set forth in claim 1, wherein step (b) includes prompting identification of said portions of said hard copy documents corresponding to said first data field.

3. A method as set forth in claim 1, wherein step (c) includes storing textual information from said portions of said hard copy documents corresponding to said first data field into said memory locations for said first data field.

4. A method as set forth in claim 1, further comprising the steps of detecting and correcting errors resulting from automated digitizing.

5. A method as set forth in claim 1, further comprising the step of utilizing a template to associate portions of said hard copy documents with specific data fields.

6. A method as set forth in claim 1, further comprising receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

7. A method as set forth in claim 1, further comprising receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a relative location on said hard copy documents containing data corresponding to a specific data field.

8. A method as set forth in claim 1, wherein step (b) includes displaying an image of a hard copy document on a display based on a contents of said memory.

9. A method as set forth in claim 8, further comprising the steps of:

recognizing portions of said hard copy documents corresponding to a second data field, said portions of said hard copy documents corresponding to said

5,258,855

17

second data field being different from said portions of said hard copy documents corresponding to said first data field; and

storing information from said portions of said hard copy documents corresponding to said second data field into memory locations or said second data field.

10. A method as set forth in claim 1, further comprising the steps of:

recognizing portions of said hard copy documents corresponding to a second data field, said portions of said hard copy documents corresponding to said second data field being different from said portions of said hard copy documents corresponding to said first data field; and

storing information from said portions of said hard copy documents corresponding to said second data field into memory locations for said second data field.

11. A method as set forth in claim 10, further comprising the step of storing image information from said portions of said hard copy documents corresponding to said second data field into said memory locations for said second data field.

12. A method as set forth in claim 10, further comprising the step of storing textual information from said portions of said hard copy documents corresponding to said second data field into said memory locations for said second data field.

13. A method as set forth in claim 10, further comprising the step of prompting identification of said portions of said hard copy documents corresponding to said second data field.

14. A method of processing information from a diversity of types of hard copy documents, said method comprising the steps of:

(a) scanning a diversity of types of hard copy documents and storing information from said diversity of types of hard copy documents into a memory, said information not fixed from one document to the next, said scanning not preceded by scanning of a separate document containing format requirements;

(b) recognizing portions of said hard copy documents corresponding to a first data field; and

(c) storing information from said portions of said hard copy documents corresponding to said first data field into memory locations for said first data field.

15. A method as set forth in claim 14, further comprising the steps of detecting and correcting errors resulting from said scanning.

16. A method as set forth in claim 14, further comprising receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a relative location on said hard copy documents containing data corresponding to a specific data field.

17. A method as set forth in claim 14, further comprising receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

18. A method as set forth in claim 14, further comprising the step of utilizing a template to associate portions of said hard copy documents with specific data fields.

18

19. A Method as set forth in claim 14, wherein step (c) includes storing textual information from said portions of said hard copy documents corresponding to said first data field into said memory locations for said first data field.

20. A method as set forth in claim 14, wherein step (b) includes prompting identification of said portions of said hard copy documents corresponding to said first data field.

21. A method as set forth in claim 14, wherein step (b) includes displaying an image of a hard copy document on a display based on a contents of said memory.

22. A method as set forth in claim 21, further comprising the steps of:

recognizing portions of said hard copy documents corresponding to a second data field, said portions of said hard copy documents corresponding to said second data field being different from said portions of said hard copy documents corresponding to said first data field; and

storing information from said portions of said hard copy documents corresponding to said second data field into memory locations for said second data field.

23. A method as set forth in claim 14, further comprising the steps of:

recognizing portions of said hard copy documents corresponding to a second data field, said portions of said hard copy documents corresponding to said second data field being different from said portions of said hard copy documents corresponding to said first data field; and

storing information from said portions of said hard copy documents corresponding to said second data field into memory locations for said second data field.

24. A method as set forth in claim 23, further comprising the step of storing image information from said portions of said hard copy documents corresponding to said second data field into said memory locations for said second data field.

25. A method as set forth in claim 23, further comprising the step of storing textual information from said portions of said hard copy documents corresponding to said second data field into said memory locations for said second data field.

26. A method as set forth in claim 23, further comprising the step of prompting identification of said portions of said hard copy documents corresponding to said second data field.

27. A method of processing data extracted from a diversity of types of hard copy documents, said method comprising the steps of:

(a) receiving output representing a diversity of types of hard copy documents from an automated digitizing unit and storing information from said diversity of types of hard copy documents into a memory as stored document information, said receiving step not preceded by scanning, via said automated digitizing unit, a separate document containing format requirements;

(b) recognizing and selecting portions of said stored document information corresponding to data fields required by an application unit;

(c) formatting said selected portions of said stored document information into a transmission format used by said application unit; and

5,258,855

19

(d) transmitting said formatted selected portions of said stored document information to said application unit.

28. A method as set forth in claim 27, further comprising the step of printing textual copies of said hard copy documents based on said stored document information.

29. A method as set forth in claim 27, wherein step (a) includes receiving output representing a diversity of types of hard copy documents from a scanner.

30. A method as set forth in claim 27, wherein step (a) includes receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a relative location on said hard copy documents containing data corresponding to a specific data field.

31. A method as set forth in claim 27, wherein step (a) includes receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

32. A method as set forth in claim 27, wherein step (a) includes the step of utilizing a template to associate portions of said hard copy documents with specific data fields.

33. A method as set forth in claim 27, further comprising detecting and correcting errors in said stored document information resulting from automated digitizing.

34. A method as set forth in claim 27, wherein step (a) includes storing textual information representing characters on said hard copy documents.

35. A method as set forth in claim 27, wherein step (a) includes storing digitized image information representing the actual appearance of said hard copy documents.

36. A method as set forth in claim 35, further comprising the step of printing copies of said hard copy documents based on said digitized image information.

37. An application program interface, comprising:
an automated digitizing unit which extracts information from a diversity of types of hard copy documents and stores said information from said diversity of types of hard copy documents in a memory as stored document information;

a processor recognizing and selecting portions of said stored document information corresponding to data fields required by an application unit, said processor operative without reference to information extracted, via said automated digitizing unit, from a separate document containing format requirements;

a formatter formatting said selected portions of said stored document information into a transmission format used by said application unit; and

an output unit transmitting said formatted selected portions of said stored document information to said application unit.

38. An interface as set forth in claim 37, wherein said automated digitizing unit includes a scanner.

39. An interface as set forth in claim 37, further comprising a search unit for searching for at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

40. An interface as set forth in claim 37, further comprising:

20

a template definition unit for defining a template which associates locations on said hard copy documents with specific data fields.

41. An interface as set forth in claim 37, further comprising an error correcting unit detecting and correcting errors resulting from extracting by said automated digitizing unit.

42. An interface as set forth in claim 37, wherein said stored document information includes textual information representing characters on said hard copy documents.

43. An interface as set forth in claim 37, wherein said stored document information includes digitized image information representing an actual appearance of said hard copy documents.

44. An interface as set forth in claim 43, further comprising a printer which prints out copies of an actual appearance of said hard copy documents based on said digitized image information.

45. A method of processing data extracted from a diversity of types of hard copy documents, said method comprising the steps of:

(a) receiving output representing a diversity of types of hard copy documents from an automated digitizing unit and storing information from said diversity of types of hard copy documents into a memory as stored document information, said receiving step not preceded by scanning, via said automated digitizing unit, of a separate document containing format requirements;

(b) recognizing and selecting portions of said stored document information corresponding to data fields required by an application unit; and

(c) transmitting said selected portions of said stored document information to said application unit.

46. A method as set forth in claim 45, wherein step (a) includes storing textual information representing characters on said hard copy documents.

47. A method as set forth in claim 45, further comprising detecting and correcting errors in said stored document information resulting from automated digitizing.

48. A method as set forth in claim 45, wherein step (a) includes the step of utilizing a template to associate portions of said hard copy documents with specific data fields.

49. A method as set forth in claim 45, wherein step (a) includes receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

50. A method as set forth in claim 45, wherein step (a) includes receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a relative location on said hard copy documents containing data corresponding to a specific data field.

51. A method as set forth in claim 45, further comprising the step of printing textual copies of said hard copy documents based on said stored document information.

52. A method as set forth in claim 45, wherein step (a) includes storing digitized image information representing an actual appearance of said hard copy documents.

53. A method as set forth in claim 52, further comprising the step of printing copies of said hard copy documents based on said digitized image information.

5,258,855

21

54. An application program interface, comprising:
an automated digitizing unit which extracts information from a diversity of types of hard copy documents and stores said information from said diversity of types of hard copy documents in a memory as stored document information;

a processor recognizing and selecting portions of said stored document information corresponding to data fields required by an application unit, said processor operative without reference to information extracted, via said automated digitizing unit, from a separate document containing format requirements; and
an output unit transmitting said selected portions of said stored document information to said application unit.

55. An application program interface as set forth in claim 54, wherein said automated digitizing unit includes a scanner.

56. An application program interface as set forth in claim 54, further comprising a search unit for searching for at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

57. An application program interface as set forth in claim 54, further comprising a template definition unit for defining a template which associates locations on said hard copy documents with specific data fields.

58. An application program interface as set forth in claim 54, further comprising an error correcting unit detecting and correcting errors resulting from extracting by said automated digitizing unit.

59. An application program interface as set forth in claim 54, wherein said stored document information includes textual information representing characters on said hard copy documents.

60. An application program interface as set forth in claim 54, wherein said stored document information includes digitized image information representing an actual appearance of said hard copy documents.

61. An application program interface as set forth in claim 60, further comprising a printer which prints out copies of an actual appearance of said hard copy documents based on said digitized image information.

62. A method of processing data extracted from a diversity of types of hard copy documents, said method comprising the steps of:

(a) receiving output representing a diversity of types of hard copy documents from an automated digitizing unit and storing information from said diversity of types of hard copy documents into a memory as stored document information, said receiving step not preceded by scanning, via said automated digitizing unit, a separate document containing format requirements;

(b) recognizing and selecting portions of said stored document information corresponding to data fields required by an application unit; and

(c) formatting said selected portions of said stored document information into a transmission format used by said application unit.

63. A method as set forth in claim 62, further comprising the step of printing textual copies of said hard copy documents based on said stored document information.

64. A method as set forth in claim 62, wherein step (a) includes storing textual information representing characters on said hard copy documents.

22

65. A method as set forth in claim 62, wherein step (a) includes storing digitized image information representing an actual appearance of said hard copy documents.

66. A method as set forth in claim 62, further comprising detecting and correcting errors in said stored document information resulting from automated digitizing.

67. A method as set forth in claim 62, wherein step (a) includes the step of utilizing a template to associate portions of said hard copy documents with specific data fields.

68. A method as set forth in claim 62, wherein step (a) includes receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

69. A method as set forth in claim 62, wherein step (a) includes receiving instructions identifying at least one character or symbol, located on said hard copy documents, which identifies a relative location on said hard copy documents containing data corresponding to a specific data field.

70. An application program interface, comprising:
an automated digitizing unit which extracts information from a diversity of types of hard copy documents and stores said information from said diversity of types of hard copy documents in a memory as stored document information;

a processor recognizing and selecting portions of said stored document information corresponding to data fields required by an application unit, said processor operative without reference to extracted information scanned, via said automated digitizing unit, of a separate document containing format requirements; and

a formatter formatting said selected portions of said stored document information into a transmission format used by said application unit.

71. An application program interface as set forth in claim 70, wherein said stored document information includes textual information representing characters on said hard copy documents.

72. An application program interface as set forth in claim 70, further comprising an error correcting unit detecting and correcting errors resulting from extracting by said automated digitizing unit.

73. An application program interface as set forth in claim 70, further comprising a template definition unit for defining a template which associates locations on said hard copy documents with specific data fields.

74. An application program interface as set forth in claim 70, further comprising a search unit for searching for at least one character or symbol, located on said hard copy documents, which identifies a location on said hard copy documents containing data corresponding to a specific data field.

75. An application program interface as set forth in claim 70, wherein said automated digitizing unit includes a scanner.

76. An application program interface as set forth in claim 70, wherein said stored document information includes digitized image information representing an actual appearance of said hard copy documents.

77. An application program interface as set forth in claim 76, further comprising a printer which prints out copies of an actual appearance of said hard copy documents based on said digitized image information.

* * * * *